Sample JSON Formats for Creating and Modifying Rules Through REST API

September 2020

This article contains sample JSON formats for POST and PUT requests for Risk Service rules.

URL to create rules: `https://rba_ui_url:ui_port/risk/config/api/v1/<tenant_id>/rules`

The following fields are mandatory for creating any Risk Service rule:

JSON Field	Type	Description
enabled	Boolean	A risk policy evaluates a rule only when it is enabled.
name	String	A unique name for the rule.
description	String	The description of the rule evaluation condition.

In this Article

External Parameters Rule JSON

```
{
  "name": "externalparamrule",
  "description": "Sample external param rule",
  "enabled": true,
  "externalParamConfigRule": [
    {
      "conditionGroup": [ {
          "condition": [ {
              "conditionName": "OS",
              "conditionValue": "win",
              "contains": true
            },
            {
              "conditionName": "patchlevel",
              "lowerThreshold": "45",
              "lessThanOrEqual": true
            }],
          "nextGroupCondition": "OR",
          "operation": "AND",
          "priority": 0
        },
        {
          "condition": [ {
              "conditionName": "gradelevel",
              "lowerThreshold": "3",
              "higherThreshold": "8",
              "lessThan": true,
              "greaterThan": true
            }],
          "nextGroupCondition": "OR",
          "operation": "AND",
          "priority": 0
        }],
      "fetchFromParamSource": true,
      "negateResult": false,
      "paramSource": [{
          "authenticationType": "None",
          "dataFormat": "JSON",
          "requestMethod": "POST",
          "requestParameter": [{
              "name": "testparam",
              "staticValue": "teststring",
              "contextValue": null
            }],
          "requestTimeout": 30000,
          "url": "http://external.site.com/rest/user/encodedUserId"
        }
      ]
    }
  ]
}
```

JSON Field	Description
conditionGroup	Collection of conditions to evaluate.
operation	Specify how multiple conditions in a condition group should be combined. For example, for ConditionGroup1, evaluate using Condition1 AND Condition2. In this sample, OS contains `win` AND patchLevel `<= 45`. Valid values are AND and `OR`.
nextGroupCondition	Specify how the condition groups should be combined. For example, ConditionGroup1 AND ConditionGroup2. Valid values are AND and `OR`.
priority	Specify the order of evaluating condition groups.
Condition Definition	
conditionName	Specify a unique name for the condition.
contains - doesNotContain - equal - equalIgnoreCase - greaterThan - greaterThanOrEqual - lessThan - lessThanOrEqual - notEqual - notEqualIgnoreCase	Set to true to enable the comparison operator. Set only one operator to true. In this sample, OS `contains` win. The only exception is for performing the `In Between` operation. For example, gradeLevel in between 3 and 8 as shown in the sample.
conditionValue	Set this field if one of the following operators is true: - contains - doesNotContain - equal - equalIgnoreCase - notEqual - notEqualIgnoreCase
lowerThreshold higherThreshold	Set these fields if one of the following operators is true: - greaterThan - greaterThanOrEqual - lessThan - lessThanOrEqual
Fetch from External source	
fetchFromParamSource	Set to true to get the data from an external source. This must be enabled for Risk service.

JSON Field	Description
url	Specify the URL of the external source to retrieve GET requests that return simple JSON responses. The conditions defined are applied on this data.
dataFormat	Specify the format of the data retrieved. Only JSON is supported.
authenticationType	Specify the type for authenticating with the parameter source URL. `Basic` and `None` are supported.
username password	Specify the credentials for authenticating with the parameter source URL. You must specify these if authenticationType is set to `Basic`.
requestMethod	Specify an HTTP method to use when invoking the parameter source URL. `GET` and `POST` are supported.
requestParameter	Specify the parameters to be sent when invoking the parameter source URL. You can specify it for POST requests.
name	Specify the request parameter name.
contextValue	Specify `tenantId`, `userId`, or `other`. If you specify `tenantId` or `userId`, the value is automatically set for the request. Specify `other` to set a custom value.
staticValue	Specify a static value to be sent for the request parameter if you specified `Other` as contextValue. For example, send the value `accessmanager` for the request param `invokedBy`.
requestTimeout	Specify a value between 1000 (1sec) and 600000 (10mins). After the specified time, the request is expired.

Cookie Rule JSON

All fields are mandatory.

```
{
  "enabled": true,
  "name": "IntranetCookieRule",
  "description": "Verify that the request has the defined cookie",
  "knownCookieRule": [
    {
      "cookieName": "cname",
      "cookieValue": "cvalue",
      "negateResult": true,
      "autoCreateCookie": true,
      "cookieMaxAge": "5",
      "cookiePath": "/test",
      "cookieSecure": "false"
    }
  ]
}
```

JSON Field	Description
cookieName	Specify the name of the cookie to be checked.
cookieValue	Specify the value that the cookie must contain.
negateResult	Specify `true` to make the rule succeed if the cookie does *not* contain the specified value.
autoCreateCookie	Specify `true` to enable creating the cookie automatically if the user authenticates successfully.
cookieMaxAge	Specify the validity of the cookie in days. The default value is one day. This is required when `autoCreateCookie` is set to true.
cookiePath	Specify the path for the cookie. This is used while auto-creating the cookie.
cookieSecure	Specify `true` if the auto-created cookie should be secure.

HTTP Header Rule JSON

```
{
  "enabled": "true",
  "name": "DeptHeaderRule",
  "description": "Validates if the request contains the finance department header",
  "httpheaderRule": [
    {
      "headerNames": [
        {
          "value": "DEPARTMENT_HEADER"
        }
      ],
      "headerCondition": [
        {
          "value": "finance"
        }
      ],
      "contains": true,
      "equals": false,
      "negateResult": false
    }
  ]
}
```

JSON Field	Description
headerNames > value	Specify the name of the header to check.
headerCondition > value	Specify the value of the header to check.
equals contains	Set any one of the following parameters to true: ◆ **equals:** The rule succeeds if the value read from the header `matches with` the value specified for `headerCondition`. ◆ **contains:** The rule succeeds if the value read from the header `contains` the value specified for `headerCondition`.
negateResult	Use this to handle negative use cases when comparing the header value with the value specified for `headerCondition`. For example, to evaluate when the value is not equal to the `headerCondition` value, set `equals: true` and `negateResult: true` To evaluate when the value does not contain the `headerCondition` value, set `contains: true` and `negateResult: true`

IP Address Rule JSON

```
{
   "enabled": "true",
   "name": "InternalNetworkRule",
   "description": "Validates if a user is logging in using the corporate network",
   "ipaddressRule": [
      {
         "ipvalue": "1.1.1.1",
         "iprange": "2.2.2.2-3.3.3.3",
         "ipsubnet": "198.51.100.0/24",
         "iplistURL": null,
         "iplistURLConnectionTimeout": 10,
         "iplistURLUpdateInterval": 300,
         "considerHistoricalData": "false",
         "negateResult": "false"
      }
   ]
}
```

Specify the fields in the **Manual IP list** section or in the **Consuming from another source** section.

JSON Field	Description
negateResult	The default value is false and the rule succeeds if a user’s IP address is in the specified list. To block users with IP addresses in the specified list, set `negateResult: true`.
considerHistoricalData	Specify true or false. When set to true, it checks the IP address in the user’s login history recorded in the database.
Manually Providing the IP Address	

JSON Field	Description
ipvalue	Specify a comma separated list of IP addresses.
iprange	Specify a comma separated list of IP address ranges.
ipsubnet	Specify the list of IP subnets that must be allowed or blocked (depending on negateResult setting).
Consuming Whitelist or Blacklist IP Addresses from Another Source	
iplistURL	Specify the URL of the source that provides the list of IP addresses to check the IP address of users.
iplistURLConnectionTimeout	Specify the value in seconds. After this time, an unresponsive connection is closed.
iplistURLUpdateInterval	Specify the value in seconds. The connection will be refreshed at the specified interval.

User Last Login Rule JSON

```
{
   "enabled": "true",
   "name": "LastLoginRule",
   "description": "Validates the last successful login of the user",
   "lastLoginCookieRule": [
      {
         "cookieMaxAge": "5",
         "cookieName": "cookieName",
         "cookiePath": "/cookiePath",
         "cookieSecure": true,
         "cryptoKey": "cryptoKey@1234",
         "lastLoginAllowedAge": "3",
         "negateResult": false
      }
   ]
}
```

JSON Field	Description
cookieName	Specify a unique name for the cookie. This rule checks if the cookie exists by using this value and determine the risk accordingly. It will also create this cookie after the successful login.
cookieMaxAge	Specify the validity of the cookie in days.
cookiePath	Specify the URL to be used in the cookie.
cookieSecure	Specify `true` if you want the cookie to be secured by HTTPS. Allowed values are `true` and `false`.
cryptoKey	Specify the crypto key to encrypt the cookie.
lastLoginAllowedAge	Specify the number of days the cookie can be accessed from the same device or system. This value must be less than the value of cookieMaxAge.

User Time of Login Rule JSON

```
{
   "enabled": "true",
   "name": "TimeOfLoginRule",
   "description": "Validates if the user is logging in during business hours",
   "userTimeOfLoginRule": [
      {
         "considerHistoricalData": false,
         "negateResult": "false",
         "dayRange": [
            {
               "fromDay": "2",
               "toDay": "6"
            }
         ],
         "timeRange": [
            {
               "fromTime": "09:00:00",
               "toTime": "17:00:00"
            }
         ]
      }
   ]
}
```

JSON Field	Description
fromDay toDay	Specify the work week: Sunday (1) to Saturday (7). In this example, it is Monday (2) to Friday (6).
fromTime toTime	Specify the working hours in a day. In this example, it is `9 AM to 5PM`.
considerHistoricalData	This is a boolean field to indicate if a user’s past login time must be considered as acceptable day and time while evaluating the rule.
negateResult	When you set it to true, the rule evaluates as true if the user is *not* logging in during the specified day and time range.

Legal Notice

For information about legal notices, trademarks, disclaimers, warranties, export and other use restrictions, U.S. Government rights, patent policy, and FIPS compliance, see https://www.microfocus.com/about/legal/.